(corresponding to US 2004/0153657A1)

(24)

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 2004-054834
(43)Date of publication of application : 19.02.2004

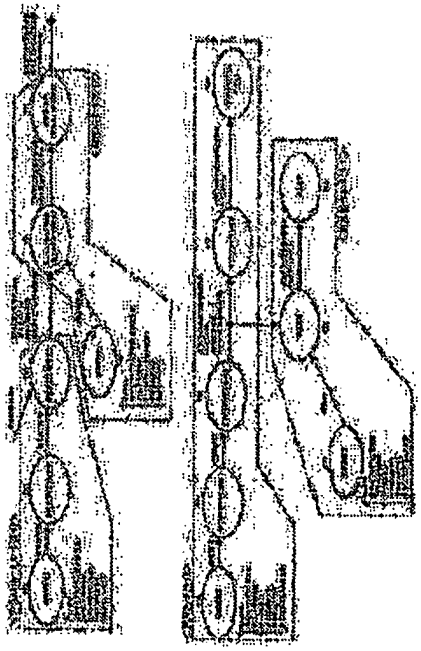
(51)Int.Cl. G06F 1/00
G09C 1/00

(21)Application number : 2002-215096
(22)Date of filing : 24.07.2002

(71)Applicant : MATSUSHITA ELECTRIC IND CO LTD
(72)Inventor : FUJIWARA MUTSUMI
NEMOTO YUSUKE
YASUI JUNICHI
MAEDA TAKUJI
ITO TAKAYUKI
YAMADA TAIJI
INOUE SHINJI

(54) PROGRAM DEVELOPMENT METHOD, PROGRAM DEVELOPMENT SUPPORT DEVICE, AND PROGRAM PACKAGING METHOD

(57)Abstract:
PROBLEM TO BE SOLVED: To provide a development environment of high security level for a key packaging system.
SOLUTION: In a system having LSIs provided with secure memories, the LSIs having a common constitution are set to a "development mode" other than a "commodity operation mode" to develop a program. They are set to an "administrator mode" to perform development and encryption of a key generation program, and they are set to a "key generation mode" to generate various keys by executing the encrypted key generation program.



(19)日本国特許庁(JP)　(12)公開特許公報(A)　(11)特許出願公開番号
特開2004-54834
(P2004-54834A)
(43)公開日　平成16年2月19日(2004.2.19)

(51)Int.Cl.[7]		F I			テーマコード（参考）
G06F	1/00	G06F	9/06	660L	5B076
G09C	1/00	G09C	1/00	660D	5J104

審査請求 未請求　請求項の数 16　OL　（全 22 頁）

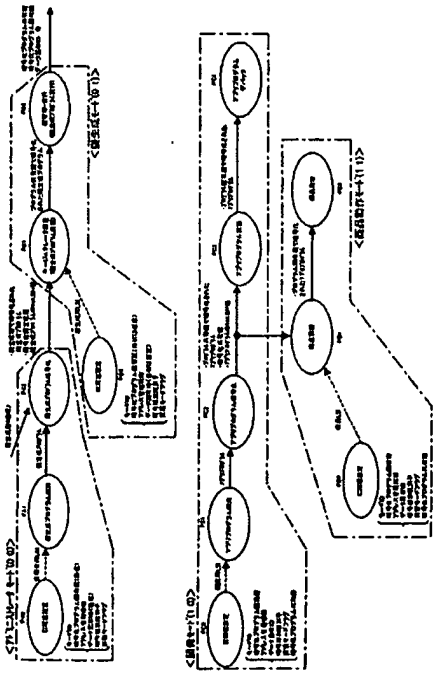
(21)出願番号　特願2002-215096(P2002-215096)
(22)出願日　平成14年7月24日(2002.7.24)

(71)出願人　000005821
松下電器産業株式会社
大阪府門真市大字門真１００６番地
(74)代理人　100077931
弁理士　前田　弘
(74)代理人　100094134
弁理士　小山　廣毅
(74)代理人　100110939
弁理士　竹内　宏
(74)代理人　100110940
弁理士　嶋田　高久
(74)代理人　100113262
弁理士　竹内　祐二
(74)代理人　100115059
弁理士　今江　克実

最終頁に続く

(54)【発明の名称】プログラム開発方法、プログラム開発支援装置およびプログラム実装方法

(57)【要約】
【課題】鍵実装システムについて、セキュリティレベルの高い開発環境を提供する。
【解決手段】セキュアメモリを備えたＬＳＩを有するシステムについて、構成が共通のＬＳＩを、<商品動作モード>とは異なる<開発モード>に設定して、プログラムの開発を行う。また、<アドミニストレータモード>に設定して鍵生成プログラムの開発および暗号化を行い、<鍵生成モード>に設定して、暗号化された鍵生成プログラムを実行させて各種の鍵を生成する。
【選択図】　図２



【特許請求の範囲】
【請求項１】
書き換え不可領域を含むセキュアメモリを備えたＬＳＩを有するシステムに、実装されるプログラムを開発する方法であって、
前記ＬＳＩと構成が共通のＬＳＩを、開発用ＬＳＩとして、プログラム実装および製品動作時における商品動作モードとは異なる、開発モードに設定する工程と、
前記開発用ＬＳＩにおいて、前記プログラムの開発を行う工程と
を備えたことを特徴とするプログラム開発方法。
【請求項２】
請求項１において、
前記ＬＳＩは、
開発モードに設定されたときは、平文プログラムを実行することができる一方、商品動作モードに設定されたときは、平文プログラムを実行することができないように、その動作が制限される
ことを特徴とするプログラム開発方法。
【請求項３】
請求項１において、
前記開発用ＬＳＩにおいて、前記プログラム開発工程で開発されたプログラムを、暗号化する暗号化工程を備えた
ことを特徴とするプログラム開発方法。
【請求項４】
請求項１において、
前記ＬＳＩは、
開発モードに設定されたとき、平文プログラムを暗号化するための鍵が生成できないように、その動作が制限される
ことを特徴とするプログラム開発方法。
【請求項５】
請求項１において、
前記ＬＳＩと構成が共通のＬＳＩを、鍵生成用ＬＳＩとして、開発モードおよび実装モードとは異なる鍵生成モードに設定する工程と、
前記鍵生成用ＬＳＩに、暗号化された鍵生成プログラムを実装し、この鍵生成プログラムを実行させることによって、鍵を生成する工程とを備えた
ことを特徴とするプログラム開発方法。
【請求項６】
請求項５において、
前記ＬＳＩは、
鍵生成モードに設定されたとき、平文プログラムを実行することができないように、その動作が制限される
ことを特徴とするプログラム開発方法。
【請求項７】
請求項５において、
前記ＬＳＩと構成が共通のＬＳＩを、管理者用ＬＳＩとして、開発モード、実装モードおよび鍵生成モードとは異なるアドミニストレータモードに設定する工程と、
前記管理者用ＬＳＩにおいて、前記鍵生成プログラムを開発し、任意の鍵で暗号化する工程とを備えた
ことを特徴とするプログラム開発方法。
【請求項８】
暗号化プログラムの開発を支援するプログラム開発支援装置であって、
前記暗号化プログラムが動作するＬＳＩと構成が共通のＬＳＩと、
平文プログラムを格納する外部メモリとを備え、

前記ＬＳＩは、
生共有鍵に係る共有鍵鍵情報を格納したセキュアメモリを備え、かつ、
前記セキュアメモリに格納された共有鍵鍵情報から、生共有鍵を得る第１のステップと、
前記外部メモリから入力された平文プログラムを、前記生共有鍵を用いて暗号化する第２のステップとが実行可能に構成されている
ことを特徴とするプログラム開発支援装置。
【請求項９】
暗号化プログラムの開発を支援するプログラム開発支援装置であって、
ＬＳＩと、平文プログラムを格納する外部メモリとを備え、
前記ＬＳＩは、
生共有鍵に係る共有鍵鍵情報を格納したセキュアメモリと、
ブートプログラムを格納するブートＲＯＭとを備え、かつ、
前記ブートＲＯＭに格納されたブートプログラムを実行することによって、
前記セキュアメモリに格納された共有鍵鍵情報から、生共有鍵を得る第１のステップと、
前記外部メモリから入力された平文プログラムを、前記生共有鍵を用いて暗号化する第２のステップとを実行する
ことを特徴とするプログラム開発支援装置。
【請求項１０】
請求項８または９において、
前記共有鍵鍵情報は、生共有鍵を生第１中間鍵で暗号化した暗号化共有鍵と、前記生第１中間鍵を生第２中間鍵で暗号化した暗号化第１中間鍵とを含むものであり、
前記第１のステップは、前記暗号化共有鍵および暗号化第１中間鍵と、プログラム暗号種とを用いて、前記生共有鍵を復号するものである
ことを特徴とするプログラム開発支援装置。
【請求項１１】
セキュアメモリを有するＬＳＩと、外部メモリとを有する鍵実装システムに、暗号化プログラムを実装する方法であって、
前記セキュアメモリに、生共有鍵に係る共有鍵鍵情報と、生固有鍵に係る固有鍵鍵情報とを格納する初期値設定処理と、
前記ＬＳＩにおいて、前記セキュアメモリに格納された共有鍵鍵情報から、生共有鍵を得る第１のステップと、
前記ＬＳＩにおいて、前記外部メモリから与えられた共有鍵暗号化プログラムを、前記第１のステップで得られた生共有鍵を用いて、復号する第２のステップと、
前記ＬＳＩにおいて、前記セキュアメモリに格納された固有鍵鍵情報から、生固有鍵を得る第３のステップと、
前記ＬＳＩにおいて、前記第２のステップで得られた平文プログラムを、前記第３のステップで得られた生固有鍵を用いて、暗号化する第４のステップとを備え、
前記第４のステップで得られた固有鍵暗号化プログラムを、前記外部メモリに実装する
ことを特徴とするプログラム実装方法。
【請求項１２】
請求項１１において、
前記ＬＳＩは、ブートプログラムを格納するブートＲＯＭを備え、
前記ブートＲＯＭに格納されたブートプログラムを前記ＬＳＩに実行させることによって、前記第１～第４のステップを実行する
ことを特徴とするプログラム実装方法。
【請求項１３】
請求項１１において、
前記固有鍵鍵情報は、前記セキュアメモリの書き換え不可領域に、格納されている
ことを特徴とするプログラム実装方法。
【請求項１４】

請求項１１において、
前記共有鍵鍵情報は、生共有鍵を生第１中間鍵で暗号化した暗号化共有鍵と、前記生第１中間鍵を生第２中間鍵で暗号化した暗号化第１中間鍵とを含むものであり、
前記第１のステップは、前記暗号化共有鍵および暗号化第１中間鍵と、プログラム暗号種とを用いて、前記生共有鍵を復号するものである
ことを特徴とするプログラム実装方法。
【請求項１５】
請求項１１において、
前記固有鍵鍵情報は、生固有鍵を生第１中間鍵で暗号化した暗号化固有鍵と、前記生第１中間鍵を生第２中間鍵で暗号化した暗号化第１中間鍵とを含むものであり、
前記第３のステップは、前記暗号化固有鍵および暗号化第１中間鍵と、プログラム暗号種とを用いて、前記生固有鍵を復号するものである
ことを特徴とするプログラム実装方法。
【請求項１６】
請求項１１において、
前記固有鍵鍵情報は、当該ＬＳＩに固有の固有ＩＤである
ことを特徴とするプログラム実装方法。
【発明の詳細な説明】
【０００１】
【発明の属する技術分野】
本発明は、鍵実装されたシステムやこれに用いるＬＳＩのプログラム開発やプログラム実装に関する技術に属する。
【０００２】
【従来の技術】
特願２００１－２８６８８１では、鍵実装システムにおいて、鍵の機密性および秘匿性を、従来よりも向上させる技術が示されている。
【０００３】
【発明が解決しようとする課題】
上述のようなシステムでは、そのプログラム開発工程や実装工程においても、いかにしてセキュリティを維持するか、ということが大きな課題となる。
【０００４】
本発明は、上述のようなシステムについて、セキュリティレベルの高い、プログラム開発の方法や環境、またはプログラム実装の方法を提案するものである。
【０００５】
【課題を解決するための手段】
請求項１の発明が講じた解決手段は、書き換え不可領域を含むセキュアメモリを備えたＬＳＩを有するシステムに実装されるプログラムを開発する方法として、前記ＬＳＩと構成が共通のＬＳＩを、開発用ＬＳＩとして、プログラム実装および製品動作時における商品動作モードとは異なる開発モードに設定する工程と、前記開発用ＬＳＩにおいて、前記プログラムの開発を行う工程とを備えたものである。
【０００６】
請求項１の発明によると、セキュアメモリを備えたＬＳＩを有するシステムに実装されるプログラムの開発が、このＬＳＩと構成が共通であり、かつ、プログラム実装および製品動作時における商品動作モードとは異なる、開発モードに設定された開発用ＬＳＩにおいて、行われる。すなわち、書き換え不可領域を含むセキュアメモリを備えており、高い秘匿性を持つＬＳＩを、その動作モードを実装モードから開発モードに変えて、プログラム開発環境として用いることによって、プログラム開発環境におけるセキュリティを、従来よりも高めることができる。
【０００７】
請求項２の発明では、請求項１におけるＬＳＩは、開発モードに設定されたときは平文プ

ログラムを実行することができる一方、商品動作モードに設定されたときは、平文プログラムを実行することができないようにその動作が制限されるものとする。

【０００８】

請求項３の発明では、前記請求項１において、前記開発用ＬＳＩにおいて、前記プログラム開発工程で開発されたプログラムを暗号化する暗号化工程を備えたものとする。

【０００９】

請求項４の発明では、前記請求項１におけるＬＳＩは、開発モードに設定されたとき、平文プログラムを暗号化するための鍵が生成できないようにその動作が制限されるものとする。

【００１０】

請求項５の発明では、前記請求項１において、前記ＬＳＩと構成が共通のＬＳＩを鍵生成用ＬＳＩとして開発モードおよび実装モードとは異なる鍵生成モードに設定する工程と、前記鍵生成用ＬＳＩに暗号化された鍵生成プログラムを実装し、この鍵生成プログラムを実行させることによって鍵を生成する工程とを備えたものとする。

【００１１】

請求項６の発明では、前記請求項５におけるＬＳＩは、鍵生成モードに設定されたとき、平文プログラムを実行することができないようにその動作が制限されるものとする。

【００１２】

請求項７の発明では、前記請求項５において、前記ＬＳＩと構成が共通のＬＳＩを管理者用ＬＳＩとして、開発モード、実装モードおよび鍵生成モードとは異なるアドミニストレータモードに設定する工程と、前記管理者用ＬＳＩにおいて前記鍵生成プログラムを開発し、任意の鍵で暗号化する工程とを備えたものとする。

【００１３】

請求項８の発明が講じた解決手段は、暗号化プログラムの開発を支援するプログラム開発支援装置として、前記暗号化プログラムが動作するＬＳＩと構成が共通のＬＳＩと、平文プログラムを格納する外部メモリとを備え、前記ＬＳＩは、生共有鍵に係る共有鍵鍵情報を格納したセキュアメモリを備え、かつ、前記セキュアメモリに格納された共有鍵鍵情報から生共有鍵を得る第１のステップと、前記外部メモリから入力された平文プログラムを前記生共有鍵を用いて暗号化する第２のステップとが実行可能に構成されているものである。

【００１４】

請求項８の発明によると、開発対象となる暗号化プログラムが動作するＬＳＩと構成が共通のＬＳＩが、開発環境として、与えられる。そして、このＬＳＩでは、セキュアメモリに格納された共有鍵鍵情報から生共有鍵が得られるとともに、この生共有鍵を用いて、外部メモリから入力された平文プログラムが暗号化される。すなわち、生共有鍵の復号と、この生共有鍵を用いた平文プログラムの暗号化とが、実行可能である。このため、プログラム開発者に生共有鍵を知られることなく、平文プログラムの暗号化を実行することができる。

【００１５】

請求項９の発明が講じた解決手段は、暗号化プログラムの開発を支援するプログラム開発支援装置として、ＬＳＩと、平文プログラムを格納する外部メモリとを備え、前記ＬＳＩは、生共有鍵に係る共有鍵鍵情報を格納したセキュアメモリと、ブートプログラムを格納するブートＲＯＭとを備え、かつ、前記ブートＲＯＭに格納されたブートプログラムを実行することによって、前記セキュアメモリに格納された共有鍵鍵情報から生共有鍵を得る第１のステップと、前記外部メモリから入力された平文プログラムを前記生共有鍵を用いて暗号化する第２のステップとを実行するものである。

【００１６】

請求項９の発明によると、ＬＳＩにおいてブートプログラムを実行することによって、セキュアメモリに格納された共有鍵鍵情報から生共有鍵が得られるとともに、外部メモリから入力された平文プログラムがこの生共有鍵を用いて暗号化される。すなわち、生共有鍵

の復号と、この生共有鍵を用いた平文プログラムの暗号化とが、外部からの指示ではなく、ブートプログラムによって実行される。このため、平文プログラムの暗号化を、プログラム開発者に生共有鍵を知られることを確実に防止しつつ、実行することができる。

【００１７】

請求項１０の発明では、請求項８または９において、前記共有鍵鍵情報は、生共有鍵を生第１中間鍵で暗号化した暗号化共有鍵と、前記生第１中間鍵を生第２中間鍵で暗号化した暗号化第１中間鍵とを含むものとし、前記第１のステップは、前記暗号化共有鍵および暗号化第１中間鍵と、プログラム暗号種とを用いて、前記生共有鍵を復号するものとする。

【００１８】

請求項１１の発明が講じた解決手段は、セキュアメモリを有するＬＳＩと外部メモリとを有する鍵実装システムに暗号化プログラムを実装する方法として、前記セキュアメモリに、生共有鍵に係る共有鍵鍵情報と、生固有鍵に係る固有鍵鍵情報とを格納する初期値設定処理と、前記ＬＳＩにおいて、前記セキュアメモリに格納された共有鍵鍵情報から生共有鍵を得る第１のステップと、前記ＬＳＩにおいて、前記外部メモリから与えられた共有鍵暗号化プログラムを前記第１のステップで得られた生共有鍵を用いて復号する第２のステップと、前記ＬＳＩにおいて、前記セキュアメモリに格納された固有鍵鍵情報から生固有鍵を得る第３のステップと、前記ＬＳＩにおいて、前記第２のステップで得られた平文プログラムを前記第３のステップで得られた生固有鍵を用いて暗号化する第４のステップとを備え、前記第４のステップで得られた固有鍵暗号化プログラムを前記外部メモリに実装するものである。

【００１９】

請求項１１の発明によると、ＬＳＩに与えられた共有鍵暗号化プログラムは、セキュアメモリに格納された共有鍵鍵情報から得られた生共有鍵を用いて、復号される。そして復号された平文プログラムは、セキュアメモリに格納された固有鍵鍵情報から得られた生固有鍵を用いて、暗号化される。すなわち、共有鍵暗号化プログラムが、暗号化する鍵が共有鍵から固有鍵に変換されて、システムに実装されることになる。このため、ユーザの持つ各製品では、互いに異なる固有鍵によって暗号化されたプログラムが実装されることになり、秘匿性が向上する。また、万一、暗号を破られた場合でも、被害を受ける製品の数が限定されることになり、従来よりもセキュリティが高まる。

【００２０】

請求項１２の発明では、請求項１１におけるＬＳＩは、ブートプログラムを格納するブートＲＯＭを備え、前記ブートＲＯＭに格納されたブートプログラムを前記ＬＳＩに実行させることによって、前記第１～第４のステップを実行するものとする。

【００２１】

請求項１３の発明では、請求項１１における固有鍵鍵情報は、前記セキュアメモリの書き換え不可領域に格納されているものとする。

【００２２】

請求項１４の発明では、請求項１１における共有鍵鍵情報は、生共有鍵を生第１中間鍵で暗号化した暗号化共有鍵と、前記生第１中間鍵を生第２中間鍵で暗号化した暗号化第１中間鍵とを含むものとし、前記第１のステップは、前記暗号化共有鍵および暗号化第１中間鍵とプログラム暗号種とを用いて前記生共有鍵を復号するものとする。

【００２３】

請求項１５の発明では、請求項１１における固有鍵鍵情報は、生固有鍵を生第１中間鍵で暗号化した暗号化固有鍵と、前記生第１中間鍵を生第２中間鍵で暗号化した暗号化第１中間鍵とを含むものとし、前記第３のステップは、前記暗号化固有鍵および暗号化第１中間鍵とプログラム暗号種とを用いて前記生固有鍵を復号するものとする。

【００２４】

請求項１６の発明では、請求項１１における固有鍵鍵情報は、当該ＬＳＩに固有の固有ＩＤであるものとする。

【００２５】

【発明の実施の形態】
以下、本発明の実施の形態について、図面を参照して説明する。なお、以下の説明では、Ｘ（鍵またはプログラム）を鍵Ｙを用いて暗号化して得た、暗号化された鍵またはプログラムのことを、Ｅｎｃ（Ｘ，Ｙ）と表すものとする。
【００２６】
図１は本実施形態に係るセキュアＬＳＩの内部構成を示すブロック図である。図１において、セキュアＬＳＩ１は外部バス１２０を介して、外部メモリ（フラッシュメモリ）１００や外部ツール１１０などと接続可能に構成されている。また、モードＩＤを与えることによって、その動作モードを設定することが可能になっている。
【００２７】
本実施形態に関わる主な構成要素について、簡単に説明する。
【００２８】
まず、セキュアＬＳＩ１は、書き換え不可領域１１を含むセキュアメモリ（セキュアＦｌａｓｈ）１０を備えている。この書き換え不可領域１１には、書き換え不可領域書き込みフラグ１２が設けられている。書き換え不可領域書き込みフラグ１２は、モードＩＤが一度セキュアメモリ１０に書き込まれると、そのフラグ値が“可”から“済”になり、それ以降の書き換え不可領域への書き込みが不能になる。なお、本実施形態では、セキュアメモリ１０および外部メモリ１００はフラッシュメモリによって構成されているが、もちろんこれに限定されるものではなく、不揮発性のメモリであればどのようなものであってもかまわない。
【００２９】
また、秘密鍵演算処理部２０は各種の鍵、およびプログラム暗号種を格納するレジスタを備えており、暗号化処理を実行する。鍵生成・更新シーケンサ３０はモードＩＤ格納レジスタ３１を有し、このモードＩＤ格納レジスタ３１に格納されているモードＩＤに応じて、秘密鍵演算処理部２０の動作、すなわち、各種の鍵を生成できるか否かを制御する。また、鍵またはプログラムがどのようなアルゴリズムや鍵長で暗号化されているかを示す暗号種別識別子を格納する暗号種別識別子格納レジスタ３２を備えている。さらに、プログラム暗号種３３が実装されている。
【００３０】
モードシーケンサ４０も、モードＩＤ格納レジスタ４１を備えており、モードＩＤ格納レジスタ４１に格納されているモードＩＤと、ジャンパー４３の値に応じて、外部ホストインターフェース（Ｉ／Ｆ）５０の動作、すなわち、外部メモリ１００に格納されたプログラムやデータをどのＩ／Ｆを介して読み込むか、を制御する。これにより、外部メモリ１００に格納された平文プログラムが実行できるか否かを制御することができる。さらに、モードシーケンサ４０は、鍵がどの手法によって暗号化されているかを示す暗号種別識別子を格納する暗号種別識別子格納レジスタ４２を備えている。
【００３１】
外部ホストＩ／Ｆ５０は、モードシーケンサ４０の制御に従って、プログラム処理部５１が有するスルー部５２、遅延部５３およびプログラム復号用暗号エンジン５４、並びに、データ処理部５５が有するスルー部５６およびコンテンツ暗号・復号用暗号エンジン５７のうちのいずれかを介して、外部メモリ１００や外部ツール１１０との間でプログラムやデータの入出力を行う。
【００３２】
ここで、後述するアドミニストレータモードを除いては、スルー部５２を介して入力されたプログラムは、セキュアＬＳＩ１内部では実行されることはない。すなわち、スルー部５２は、平文プログラムの暗号化、またはすでに暗号化されたプログラムを別の鍵を用いて再暗号化するときに有効とされるものであり、セキュアＬＳＩ１は、後述するアドミニストレータモードを除いては、スルー部５２を介して入力されたプログラムへは動作を遷移しないように構成されている。したがって、例えば商品となったセキュアＬＳＩ１はスルー部５２を介して平文プログラムを取り込んだとしても、これを実行することはできな

い。なお、平文プログラムを実行するときは、セキュアＬＳＩ１は遅延部５３を介してプログラムをその内部に入力する。

【００３３】

ブートＲＯＭ６０は、セキュアＬＳＩ１の起動動作を制御するブートプログラムを格納している。ＨＡＳＨ演算部７０は、セキュアＬＳＩ１に読み込まれたプログラムについてその正当性を検証するために、ＨＡＳＨ値を演算する。

【００３４】

また、外部メモリ１００には、プログラムやコンテンツが格納されている。外部ツール１１０には、セキュアＬＳＩ１の最初の起動時にセキュアメモリ１０に格納する各種の初期値が格納されている。この初期値の種類は、設定される動作モードに応じて、異なったものになる。

【００３５】

図２は図１のセキュアＬＳＩ１を用いた開発および製品化の全体の流れを表す図である。図２に示すように、セキュアＬＳＩ１は、アドミニストレータモード（モードＩＤ：００）、鍵生成モード（モードＩＤ：０１）、開発モード（モードＩＤ：１０）および商品動作モード（モードＩＤ：１１）の４種類の動作モードで、動作する。

【００３６】

まず、アドミニストレータモードに設定されたセキュアＬＳＩ１は、管理者用ＬＳＩとして、動作する。管理者用ＬＳＩでは、鍵生成プログラムが開発され、また、その鍵生成プログラムが任意の鍵生成鍵を用いて暗号化される。

【００３７】

鍵生成モードに設定されたセキュアＬＳＩ１は、鍵生成用ＬＳＩとして、動作する。鍵生成用ＬＳＩでは、管理者用ＬＳＩにおいて生成された、暗号化された鍵生成プログラムが実装され、この鍵生成プログラムを実行することによって、各種の鍵が生成される。

【００３８】

開発モードに設定されたセキュアＬＳＩ１は、開発用ＬＳＩとして、動作する。開発用ＬＳＩでは、実際の製品で実行されるアプリケーション用プログラムが開発される。そして、このアプリケーション用プログラムが、プログラム共有鍵を用いて暗号化される。

【００３９】

商品動作モードに設定されたセキュアＬＳＩ１は、実際の商品ＬＳＩとして、動作する。商品ＬＳＩでは、開発用ＬＳＩにおいて生成された、プログラム共有鍵で暗号化されたアプリケーション用プログラムが実装され、その内部で、プログラム固有鍵で暗号化されたアプリケーション用プログラムに、変換される。なお、この変換処理は、開発用ＬＳＩでも、アプリケーション用プログラムのデバッグのために、実行可能になっている。

【００４０】

以下、各モードにおけるセキュアＬＳＩ１の動作の詳細について、フローチャートおよびデータフローを参照して、説明する。セキュアＬＳＩ１は、ブートＲＯＭ６０に格納されたブートプログラムを実行することによって、以下のような動作を行う。

【００４１】

図３はブートプログラムの全体的な処理の流れを示すフローチャートである。セキュアＬＳＩ１に電源が投入されると、ブートＲＯＭ６０に格納されたブートプログラムがＣＰＵ６５によって実行される。図３に示すように、まず、各ハードウェアを初期化する（ＳＺ０）。そして、外部ツール１１０からさまざまな初期値を読み込み、セキュアメモリ１０に設定する（ＳＺ１）。

【００４２】

図２２は初期値設定処理ＳＺ１のフローチャートである。まず、ジャンパー４４で、セキュアメモリ１０がＬＳＩ内に実装されているか否かの判定を行う。次に、書き換え不可領域書き込みフラグ１２が“済”であるか否かを判定し、“済”であるときはすでにセキュアメモリ１０に初期値が設定されているので、処理ＳＺ１を終了する。書き換え不可領域書き込みフラグ１２が“可”であるときは、セキュアメモリ１０に初期値を書き込んでい

く。モードＩＤに加えて、暗号化されたプログラム固有鍵、アドレス管理情報、データ固有鍵をセキュアメモリ１０の書き換え不可領域１１に書き込む。なお、最初の判定の結果、セキュアメモリ１０がＬＳＩの外部にあると判定されたときは、モードＩＤは商品動作モードを表す値に上書きされる。これにより、セキュアメモリ１０がＬＳＩパッケージ外にあるような不正な製品は、商品動作モードでしか動作できない。

【００４３】

次に、書き込み不可領域書き込みフラグ１２を“済”にセットする。これによって、以後の書き換え不可領域１１の書き換えはできなくなる。さらに、通常領域１３，１４に暗号種別識別子および実装モードフラグを書き込む。そして、モードＩＤがアドミニストレータモード以外のモードを示すときは、これらに加えて、暗号化された共有鍵／鍵生成鍵も通常領域１３，１４に書き込む。

【００４４】

その後、前処理ＳＺ２を実行する。図４は前処理ＳＺ２のデータフローである。ここでは、セキュアメモリ１０の書き込み不可領域１１に設定されたモードＩＤが、鍵生成・更新シーケンサ３０のモードＩＤ格納レジスタ３１と、モードシーケンサ４０のモードＩＤ格納レジスタ４１とに設定される。また、セキュアメモリ１０の第１の通常領域１３に設定された暗号種別識別子が、鍵生成・更新シーケンサ３０の暗号種別識別子格納レジスタ３２と、モードシーケンサ４０の暗号種別識別子格納レジスタ４２とに設定される。さらに、セキュアメモリ１０の書き換え不可領域１１に格納されたアドレス管理情報が、ＭＥＭＣ８０の暗号アドレス区分格納レジスタ８１に設定される。ここまでの動作は、図２における初期値設定フェーズＰＡ０，ＰＢ０，ＰＣ０，ＰＤ０に対応している。

【００４５】

その後は、モードＩＤの値に応じて、それぞれのモードにおける動作を行う（ＳＺ３）。

【００４６】

＜アドミニストレータモード＞

モードＩＤが「００」のとき、セキュアＬＳＩ１はアドミニストレータモードになり、ジャンパー４３の値に応じて（ＳＡ０）、平文プログラム実行処理ＳＡ１、またはプログラム暗号化処理ＳＡ２を実行する。

【００４７】

鍵生成プログラム開発フェーズＰＡ１では、平文プログラム実行処理ＳＡ１が行われ、ここで、鍵生成プログラムが生成される。この鍵生成プログラムは外部メモリ１００に格納される。

【００４８】

鍵生成プログラム暗号化フェーズＰＡ２では、まず、図５のデータフローのように、鍵生成プログラムを実行させることによって、与えられた任意の鍵生成鍵を暗号化する。すなわち、外部ホストＩ／Ｆ５０では、モードシーケンサ４０によって、プログラム処理部５１のスルー部５２が有効化される。そして、外部メモリ１００に格納された鍵生成プログラムが、スルー部５２を介してＣＰＵ６５に与えられ、実行される。この鍵生成プログラムを実行することによって、外部メモリ１００に格納された鍵生成鍵が、秘密鍵演算処理部２０によって、鍵生成・更新シーケンサ３０に実装されたプログラム暗号種を用いて暗号化される。

【００４９】

なお、本実施形態では、鍵の暗号化は、第１中間鍵と第２中間鍵とを用いて行う。すなわち、暗号化の結果、平文鍵（ここでは鍵生成鍵）を第１中間鍵（ここではＭＫ１）で暗号化した暗号化鍵（ここではＥｎｃ（鍵生成鍵、ＭＫ１））と、第１中間鍵を第２中間鍵（ここではＣＫ）で暗号化した暗号化第１中間鍵（ここではＥｎｃ（ＭＫ１，ＣＫ））とが得られる。もちろん、本発明は、このような鍵の暗号化手法に限定されるものではない。

【００５０】

その後、プログラム暗号化処理ＳＡ２が実行される。図６はこのプログラム暗号化処理ＳＡ２のフローチャート、図７はデータフローである。まず、外部メモリ１００に格納され

ている、暗号化された鍵生成鍵Ｅｎｃ（鍵生成鍵，ＭＫ１），Ｅｎｃ（ＭＫ１，ＣＫ）を、外部ホストＩ／Ｆ５０のスルー部５２を介して、秘密鍵演算処理部２０に設定する（ＳＡ２１）。そして、この暗号化された鍵生成鍵を、鍵生成・更新シーケンサ３０に実装されたプログラム暗号種を用いて復号し、鍵生成鍵を得る（ＳＡ２２）。その後、外部メモリ１００に格納されていた平文の鍵生成プログラムを取り込み、これをＳＡ２２で復号した鍵生成鍵を用いて暗号化し、外部メモリ１００に書き込む（ＳＡ２３）。さらに、外部メモリ１００の平文の鍵生成プログラムに対して、ＨＡＳＨ演算部７０によってＨＡＳＨ演算を行い、算出したＨＡＳＨ値を外部メモリ１００に書き込む（ＳＡ２４）。

【００５１】

このような動作によって、アドミニストレータモードでは、鍵生成鍵で暗号された鍵生成プログラムＥｎｃ（鍵生成プログラム，鍵生成鍵）と、暗号化された鍵生成鍵Ｅｎｃ（鍵生成鍵，ＭＫ１），Ｅｎｃ（ＭＫ１，ＣＫ）と、鍵生成プログラムのＨＡＳＨ値とが、生成される。

【００５２】

＜鍵生成モード＞

モードＩＤが「０１」のとき、セキュアＬＳＩ１は鍵生成モードになり、実装モードフラグの値に応じて（ＳＢ０）、キージェネレータ製造処理ＳＢ１、または鍵管理・発行処理ＳＢ２を実行する。

【００５３】

キージェネレータ製造フェーズＰＢ１では、キージェネレータ製造処理ＳＢ１が実行される。図８はこの処理ＳＢ１のフローチャート、図９および図１０はデータフローである。ここでは、モードＩＤと実装モードフラグの値によって、外部ホストＩ／Ｆ部５０が有するプログラム処理部５１においてスルー部５２が有効に設定されている。

【００５４】

まず、セキュアメモリ１０の書き込み不可領域１１に格納されている、暗号化されたプログラム固有鍵Ｅｎｃ（プログラム固有鍵、ＭＫ０），Ｅｎｃ（ＭＫ０，ＣＫ）を秘密鍵演算処理部２０の暗号鍵格納レジスタに設定する（ＳＢ１１）。そして、この暗号化されたプログラム固有鍵を、鍵生成・更新シーケンサ３０に実装されたプログラム暗号種を用いて復号し、プログラム固有鍵を得る（ＳＢ１２）。次に、初期値設定フェーズＰＢ０において設定された，暗号化された鍵生成鍵Ｅｎｃ（鍵生成鍵，ＭＫ１），Ｅｎｃ（ＭＫ１，ＣＫ）を秘密鍵演算処理部２０の暗号鍵格納レジスタに設定し（ＳＢ１３）、この暗号化された鍵生成鍵を、鍵生成・更新シーケンサ３０に実装されたプログラム暗号種を用いて復号し、鍵生成鍵を得る（ＳＢ１４）。

【００５５】

その後、外部メモリ１００に格納されていた，鍵生成鍵で暗号化された鍵生成プログラムＥｎｃ（鍵生成プログラム，鍵生成鍵）を、外部ホストＩ／Ｆ５０が有するプログラム処理部５１のスルー部５２を介して、秘密鍵演算処理部２０に取り込む（ＳＢ１５）。そして、取り込んだ暗号化された鍵生成プログラムを、鍵生成鍵で復号した後、プログラム固有鍵で暗号化し、暗号化された鍵生成プログラムＥｎｃ（鍵生成プログラム，プログラム固有鍵）を得る（ＳＢ１６）。そして、外部メモリ１００に書き込む（ＳＢ１７）。次に、外部メモリ１００に格納されていたＨＡＳＨ値を、スルー部５２を介して、セキュアメモリ１０の通常領域１３に設定する（ＳＢ１８）。

【００５６】

また、セキュアメモリ１０の通常領域１３に格納された実装モードフラグの値を、ＣＰＵ６５によって“ＯＦＦ”に設定する（ＳＢ１９）。そして、セキュアメモリ１０の通常領域１３に格納されている，暗号化された鍵生成鍵Ｅｎｃ（鍵生成鍵，ＭＫ１），Ｅｎｃ（ＭＫ１，ＣＫ）を削除する（ＳＢ１Ａ）とともに、外部メモリ１００に格納されていた，暗号化された鍵生成プログラムＥｎｃ（鍵生成鍵プログラム，鍵生成鍵）およびＨＡＳＨ値を削除する（ＳＢ１Ｂ）。

【００５７】

鍵管理・発行フェーズＰＢ２では、鍵管理・発行処理ＳＢ２が実行される。図１１はこの処理ＳＢ２のフローチャート、図１２および図１３はデータフローである。ここでは、モードＩＤと実装モードフラグの値によって、外部ホストＩ／Ｆ部５０が有するプログラム復号用暗号エンジン５４が有効に設定されている。

【００５８】

まず、セキュアメモリ１０の書き込み不可領域１１に格納されている、暗号化されたプログラム固有鍵Ｅｎｃ（プログラム固有鍵、ＭＫ０），Ｅｎｃ（ＭＫ０，ＣＫ）を秘密鍵演算処理部２０の暗号鍵格納レジスタに設定する（ＳＢ２１）。そして、この暗号化されたプログラム固有鍵を、鍵生成・更新シーケンサ３０に実装されたプログラム暗号種を用いて復号し、プログラム固有鍵を得る（ＳＢ２２）。得たプログラム固有鍵は、外部ホストＩ／Ｆ５０のプログラム復号用暗号エンジン５４のプログラム固有鍵格納レジスタに設定される（ＳＢ２３）。

【００５９】

その後、外部メモリ１００に格納されていた，プログラム固有鍵で暗号化された鍵生成プログラムＥｎｃ（鍵生成プログラム，プログラム固有鍵）を、外部ホストＩ／Ｆ５０が有するプログラム処理部５１のプログラム復号用暗号エンジン５４を介して復号し、ＨＡＳＨ演算部７０に取り込み、ＨＡＳＨ値を演算する（ＳＢ２４）。そして、この演算したＨＡＳＨ値と、セキュアメモリ１０の通常領域１３に格納されていたＨＡＳＨ値とを比較し、鍵生成プログラムが改ざんされていないかどうかをチェックする（ＳＢ２５）。ＨＡＳＨ値が一致していたとき（ＳＢ２６でＮｏ）、外部メモリ１００に格納されていた鍵生成プログラムＥｎｃ（鍵生成プログラム，プログラム固有鍵）に処理を遷移し、鍵の生成を実行する（ＳＢ２７）。一方、ＨＡＳＨ値が一致していないとき（ＳＢ２６でＹｅｓ）は、何らかの不正が行われたものと推定して、不正アクセス時制御による処理を実行する（ＳＢ２８）。

【００６０】

鍵生成モードにおいては、スルー部５２を有効にしてプログラムを入力する、またはプログラム復号用暗号エンジン５４を有効にして暗号化されたプログラムを復号して入力するのみであるので、平文プログラムを実行することができないように、セキュアＬＳＩ１の動作が制限される。

【００６１】

＜開発モード＞

モードＩＤが「１０」のとき、セキュアＬＳＩ１は開発モードになり、ジャンパー４３の値に応じて（ＳＣ０）、プログラム暗号化処理ＳＣ１、平文プログラム実行処理ＳＣ２、プログラム実装処理ＳＣ３、または暗号化プログラム実行処理ＳＣ４を実行する。

【００６２】

アプリケーションプログラム開発フェーズＰＣ１では、遅延部５３を有効として、平文プログラム実行処理ＳＣ２が行われ、アプリケーションプログラムが開発される。開発されたアプリケーションプログラムは、外部メモリ１００に格納される。

【００６３】

アプリケーションプログラム暗号化フェーズＰＣ２では、プログラム暗号化処理ＳＣ１が実行される。図１４はこのプログラム暗号化処理ＳＣ１のフローチャート、図１５はデータフローである。まず、セキュアメモリ１０の通常領域１４に格納された共有鍵鍵情報としての暗号化されたプログラム共有鍵Ｅｎｃ（プログラム共有鍵，ＭＫ２），Ｅｎｃ（ＭＫ２，ＣＫ）を秘密鍵演算処理部２０に設定する（ＳＣ１１）。そして、この暗号化されたプログラム共有鍵を、鍵生成・更新シーケンサ３０に実装されたプログラム暗号種を用いて復号し、プログラム共有鍵を得る（ＳＣ１２）。その後、外部メモリ１００に格納された平文のアプリケーションプログラムを取り込み、これをＳＣ１２で復号したプログラム共有鍵を用いて暗号化し、外部メモリ１００に書き込む（ＳＣ１３）。さらに、外部メモリ１００の平文のアプリケーションプログラムに対して、ＨＡＳＨ演算部７０によってＨＡＳＨ演算を行い、算出したＨＡＳＨ値を外部メモリ１００に書き込む（ＳＣ１４）。

【００６４】
このような動作によって、プログラム共有鍵で暗号されたアプリケーションプログラムＥｎｃ（アプリケーションプログラム，プログラム共有鍵）と、アプリケーションプログラムのＨＡＳＨ値とが、生成される。
【００６５】
次に、アプリケーションプログラム実装フェーズＰＣ３では、プログラム実装処理ＳＣ３が実行され、アプリケーションプログラムデバッグフェーズＰＣ４では、暗号化プログラム実行処理ＳＣ４が実行される。これらの処理は、商品動作モードにおける各処理ＳＤ１，ＳＤ２と同様であるので、詳細は後述する。
【００６６】
このように、書き換え不可領域１１を含むセキュアメモリ１０を有し、高い秘匿性を持つＬＳＩ１を、その動作モードを実装モードから開発モードに変えてプログラム開発環境として用いることによって、プログラム開発環境におけるセキュリティを、従来よりも高めることができる。
【００６７】
また、セキュアメモリ１０に格納された共有鍵鍵情報としての暗号化された共有鍵から生共有鍵が復号され、そしてこの生共有鍵を用いて平文プログラムの暗号化が実行されるので、プログラム開発者に生共有鍵を知られることなく、平文プログラムの暗号化を実行することができる。
【００６８】
また、生共有鍵の復号と、この生共有鍵を用いた平文プログラムの暗号化とが、外部からの指示ではなく、ブートプログラムによって実行されるので、平文プログラムの暗号化を、プログラム開発者に生共有鍵を知られることを確実に防止しつつ、実行することができる。
【００６９】
＜商品動作モード＞
モードＩＤが「１１」のとき、セキュアＬＳＩ１は商品動作モードになり、実装モードフラグの値に応じて（ＳＤ０）、プログラム実装処理ＳＤ１、または通常ブート処理ＳＤ２を実行する。
【００７０】
商品実装フェーズＰＤ１では、プログラム実装処理ＳＤ１が実行される。図１６はこの処理ＳＤ１のフローチャート、図１７および図１８はデータフローである。ここでは、モードＩＤと実装モードフラグの値によって、外部ホストＩ／Ｆ部５０が有するプログラム処理部５１においてスルー部５２が有効に設定されている。
【００７１】
まず、セキュアメモリ１０の書き込み不可領域１１に格納された、固有鍵鍵情報としての暗号化されたプログラム固有鍵（プログラム固有鍵、ＭＫ０），Ｅｎｃ（ＭＫ０，ＣＫ）を秘密鍵演算処理部２０の暗号鍵格納レジスタに設定する（ＳＤ１１）。そして、この暗号化されたプログラム固有鍵を、鍵生成・更新シーケンサ３０に実装されたプログラム暗号種を用いて復号し、プログラム固有鍵を得る（ＳＤ１２）。次に、初期値設定フェーズＰＤ０において設定された，共有鍵鍵情報としての暗号化されたプログラム共有鍵Ｅｎｃ（プログラム共有鍵，ＭＫ２），Ｅｎｃ（ＭＫ２，ＣＫ）を秘密鍵演算処理部２０の暗号鍵格納レジスタに設定し（ＳＤ１３）、この暗号化されたプログラム共有鍵を、鍵生成・更新シーケンサ３０に実装されたプログラム暗号種を用いて復号し、プログラム共有鍵を得る（ＳＤ１４）。
【００７２】
その後、外部メモリ１００に格納されていた，プログラム共有鍵で暗号化されたアプリケーションプログラムＥｎｃ（アプリケーションプログラム，プログラム共有鍵）を、外部ホストＩ／Ｆ５０が有するプログラム処理部５１のスルー部５２を介して、秘密鍵演算処理部２０に取り込む（ＳＤ１５）。そして、取り込んだ暗号化されたアプリケーションプ

ログラムを、プログラム共有鍵で復号した後、プログラム固有鍵で暗号化し、暗号化されたアプリケーションプログラムＥｎｃ（アプリケーションプログラム，プログラム固有鍵）を得る（ＳＤ１６）。そして、外部メモリ１００に書き込む（ＳＤ１７）。次に、外部メモリ１００に格納されていたＨＡＳＨ値を、スルー部５２を介して、セキュアメモリ１０の通常領域１３に設定する（ＳＤ１８）。

【００７３】

また、セキュアメモリ１０の通常領域１３に格納された実装モードフラグの値を、ＣＰＵ６５によって“ＯＦＦ”に設定する（ＳＤ１９）。そして、セキュアメモリ１０の通常領域１３に格納されている，暗号化されたプログラム共有鍵Ｅｎｃ（プログラム共有鍵，ＭＫ１），Ｅｎｃ（ＭＫ１，ＣＫ）を削除する（ＳＤ１Ａ）とともに、外部メモリ１００に格納されていた，暗号化されたアプリケーションプログラムＥｎｃ（アプリケーションプログラム，プログラム共有鍵）およびＨＡＳＨ値を削除する（ＳＤ１Ｂ）。

【００７４】

すなわち、共有鍵暗号化プログラムが、暗号化する鍵が共有鍵から固有鍵に変換されて、システムに実装されることになる。このため、ユーザの持つ各製品では、互いに異なる固有鍵によって暗号化されたプログラムが実装されることになり、秘匿性が向上する。また、万一、暗号を破られた場合でも、被害を受ける製品の数が限定されることになり、従来よりもセキュリティが高まる。

【００７５】

なお、固有鍵の生成は、固有ＩＤを基にして行ってもよい。すなわち、セキュアＬＳＩ１毎に、個別の固有ＩＤを固有鍵鍵情報としてセキュアメモリ１０に実装しておき、この商品実装フェーズＰＤ１において、ブートプログラムによって、実装された固有ＩＤから固有鍵を生成するようにしてもよい。

【００７６】

商品動作フェーズＰＤ２では、通常ブート処理ＳＤ２が実行される。図１９はこの処理ＳＤ２のフローチャート、図２０および図２１はデータフローである。ここでは、モードＩＤと実装モードフラグの値によって、外部ホストＩ／Ｆ部５０が有するプログラム復号用暗号エンジン５４が有効に設定されている。

【００７７】

まず、セキュアメモリ１０の書き込み不可領域１１に格納されている、暗号化されたプログラム固有鍵Ｅｎｃ（プログラム固有鍵、ＭＫ０），Ｅｎｃ（ＭＫ０，ＣＫ）を秘密鍵演算処理部２０の暗号鍵格納レジスタに設定する（ＳＤ２１）。そして、この暗号化されたプログラム固有鍵を、鍵生成・更新シーケンサ３０に実装されたプログラム暗号種を用いて復号し、プログラム固有鍵を得る（ＳＤ２２）。得たプログラム固有鍵は、外部ホストＩ／Ｆ５０のプログラム復号用暗号エンジン５４のプログラム固有鍵格納レジスタに設定する（ＳＤ２３）。

【００７８】

その後、セキュアメモリ１０の書き込み不可領域１１に格納されているデータ固有ＩＤを秘密鍵演算処理部２０の固有ＩＤ格納レジスタに設定する（ＳＤ２４）。また、ＣＰＵ６５によって乱数を生成し、秘密鍵演算処理部２０の乱数格納レジスタに設定する（ＳＤ２５）。そして、秘密鍵演算処理部２０によって、データ固有ＩＤと乱数からデータ固有鍵を生成する（ＳＤ２６）。

【００７９】

その後、外部メモリ１００に格納されていた，プログラム固有鍵で暗号化されたアプリケーションプログラムＥｎｃ（アプリケーションプログラム，プログラム固有鍵）を、外部ホストＩ／Ｆ５０が有するプログラム処理部５１のプログラム復号用暗号エンジン５４を介して復号し、ＨＡＳＨ演算部７０に取り込み、ＨＡＳＨ値を演算する（ＳＤ２７）。そして、この演算したＨＡＳＨ値と、セキュアメモリ１０の通常領域１３に格納されていたＨＡＳＨ値とを比較し、アプリケーションプログラムが改ざんされていないかどうかをチェックする（ＳＤ２８）。ＨＡＳＨ値が一致していたとき（ＳＤ２９でＮｏ）、外部メモ

リ１００に格納されていたアプリケーションプログラムＥｎｃ（アプリケーションプログラム，プログラム固有鍵）に処理を遷移し、アプリケーションを実行する（ＳＤ２Ａ）。一方、ＨＡＳＨ値が一致していないとき（ＳＤ２９でＹｅｓ）は、何らかの不正が行われたものと推定して、不正アクセス時制御による処理を実行する（ＳＤ２Ｂ）。

【００８０】

商品動作モードにおいては、スルー部５２を有効にしてプログラムを入力する、またはプログラム復号用暗号エンジン５４を有効にして暗号化されたプログラムを復号して入力するのみであるので、平文プログラムを実行することができないように、セキュアＬＳＩ１の動作が制限される。

【００８１】

なお、開発モードおよび商品動作モードにおいては、外部から、秘密鍵演算処理部２０を用いて鍵を生成する処理を実行させようとしても、鍵生成・更新シーケンサ３０によって判別されて実行されない。すなわち、鍵生成・更新シーケンサ３０は、開発モードおよび商品動作モードにおいては、起動時以外にプログラム暗号種を用いることができないように動作を制限するので、鍵を生成する処理は実行することができない。

【００８２】

なお、本実施形態では、外部メモリ１００にプログラムやデータが格納されており、外部ツール１１０に、セキュアメモリ１０に実装される初期値が格納されているが、これらはどちらに格納されていてもかまわない。例えば、プログラムやデータが外部ツール１１０から読み込まれ、再暗号化されたとしても何ら問題はない。

【００８３】

なお、本実施形態では、ブートプログラムによって、各処理を実行するものとしたが、本発明はこれに限られるものではなく、処理の一部または全部を、他の手段によって実行してもかまわない。ただし、外部からの指示ではなく、ブートプログラムによって処理を実行させることによって、セキュリティをより高めることができる。

【００８４】

【発明の効果】

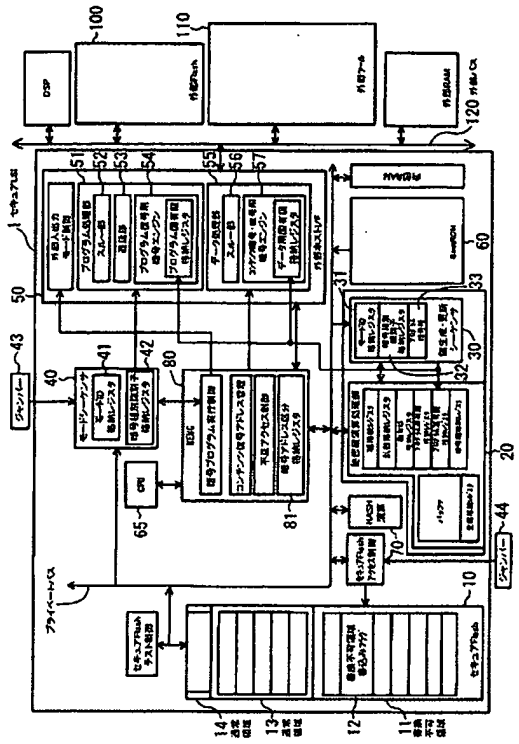
以上のように本発明によると、書き換え不可領域を含むセキュアメモリを有し、高い秘匿性を持つＬＳＩを、その動作モードを実装モードから開発モードに変えて、プログラム開発環境として用いることによって、プログラム開発環境におけるセキュリティを、従来よりも高めることができる。

【図面の簡単な説明】

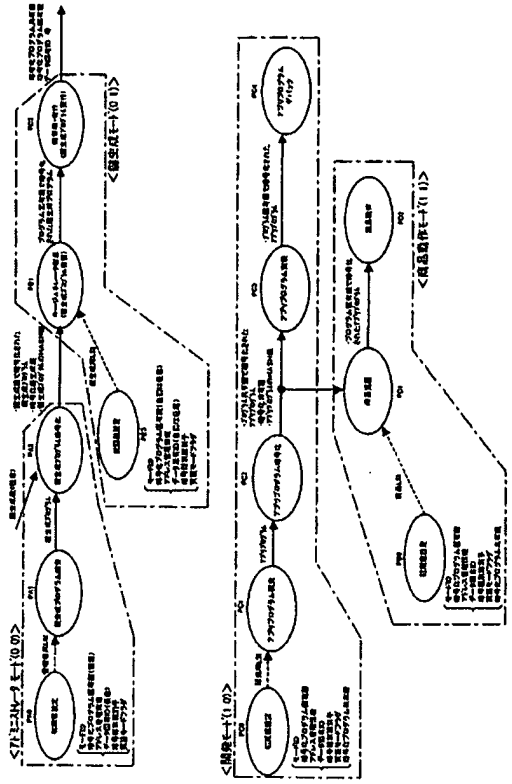
【図１】本発明の実施形態に係るセキュアＬＳＩの構成を示すブロック図である。
【図２】図１のセキュアＬＳＩを用いた開発および製品化の全体の流れを表す図である。
【図３】ブートプログラムの全体的な処理の流れを示すフローチャートである。
【図４】前処理ＳＺ２のデータフローである。
【図５】鍵生成鍵の暗号化のデータフローである。
【図６】プログラム暗号化処理ＳＡ２のフローチャートである。
【図７】プログラム暗号化処理ＳＡ２のデータフローである。
【図８】鍵生成モードにおけるキージェネレータ製造処理ＳＢ１のフローチャートである。
【図９】キージェネレータ製造処理ＳＢ１のデータフローである。
【図１０】キージェネレータ製造処理ＳＢ１のデータフローである。
【図１１】鍵生成モードにおける鍵管理・発行処理ＳＢ２のフローチャートである。
【図１２】鍵管理・発行処理ＳＢ２のデータフローである。
【図１３】鍵管理・発行処理ＳＢ２のデータフローである。
【図１４】開発モードにおけるプログラム暗号化処理ＳＣ１のフローチャートである。
【図１５】プログラム暗号化処理ＳＣ１のデータフローである。
【図１６】商品動作モードにおけるプログラム実装処理ＳＤ１のフローチャートである。
【図１７】プログラム実装処理ＳＤ１のデータフローである。

【図１８】プログラム実装処理ＳＤ１のデータフローである。
【図１９】商品動作モードにおける通常ブート処理ＳＤ２のフローチャートである。
【図２０】通常ブート処理ＳＤ２のデータフローである。
【図２１】通常ブート処理ＳＤ２のデータフローである。
【図２２】初期値設定処理ＳＺ１のフローチャートである。
【符号の説明】
１０　セキュアメモリ
１１　書き換え不可領域
１２　書き換え不可領域書き込みフラグ
２０　秘密鍵演算処理部
３０　鍵生成・更新シーケンサ
３１　モードＩＤ格納レジスタ
３２　暗号種別識別子格納レジスタ
３３　プログラム暗号種
４０　モードシーケンサ
４１　モードＩＤ格納レジスタ
４２　暗号種別識別子格納レジスタ
５０　外部ホストＩ／Ｆ
５１　プログラム処理部
５２　スルー部
５３　遅延部
５４　プログラム復号用暗号エンジン
５５　データ処理部
５６　スルー部
５７　コンテンツ暗号・復号用暗号エンジン
６０　ブートＲＯＭ
７０　ＨＡＳＨ演算部
１００　外部メモリ

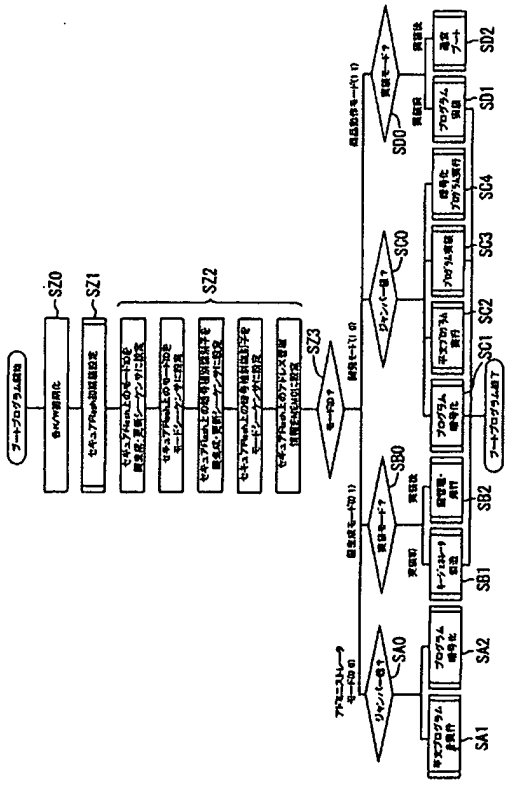
【図１】



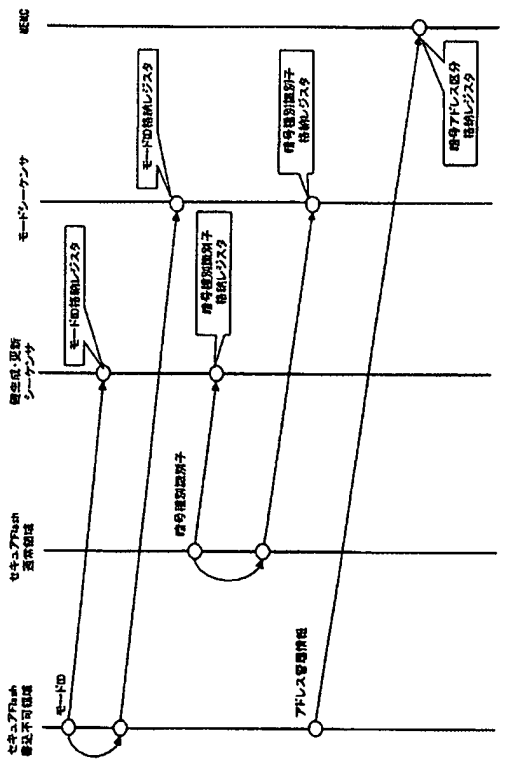
【図２】



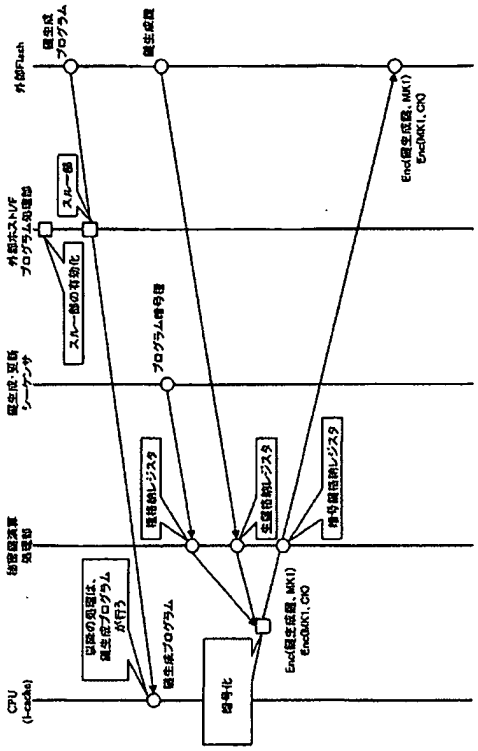
【図３】



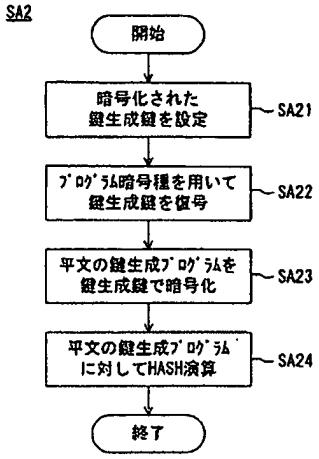
【図４】



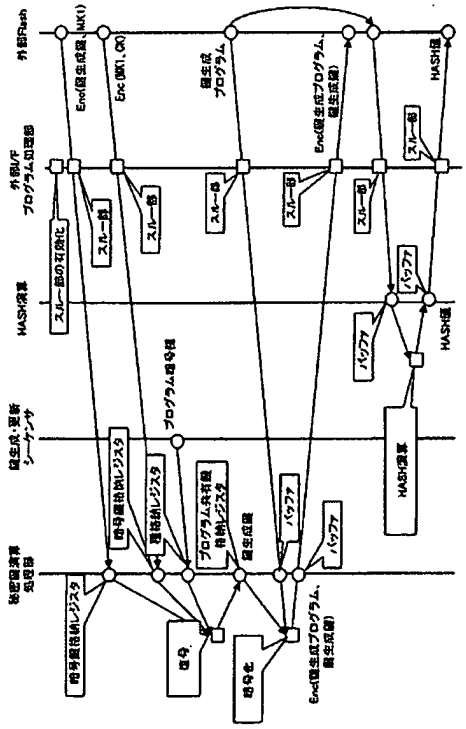
【図５】



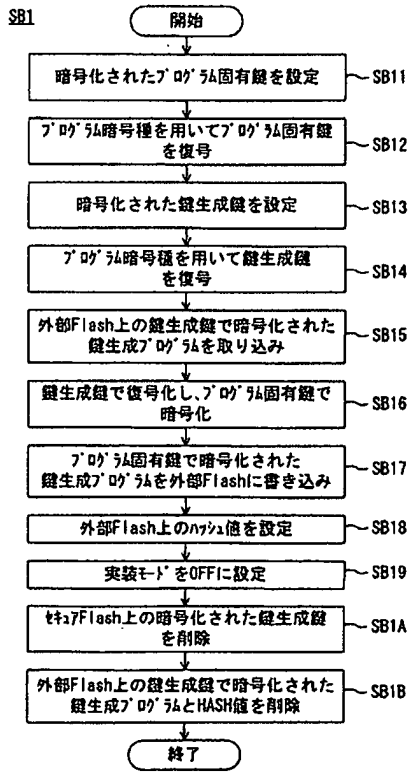
【図６】


SA2
開始
暗号化された
鍵生成鍵を設定
SA21
ﾌﾟﾛｸﾞﾗﾑ暗号種を用いて
鍵生成鍵を復号
SA22
平文の鍵生成ﾌﾟﾛｸﾞﾗﾑを
鍵生成鍵で暗号化
SA23
平文の鍵生成ﾌﾟﾛｸﾞﾗﾑ
に対してHASH演算
SA24
終了

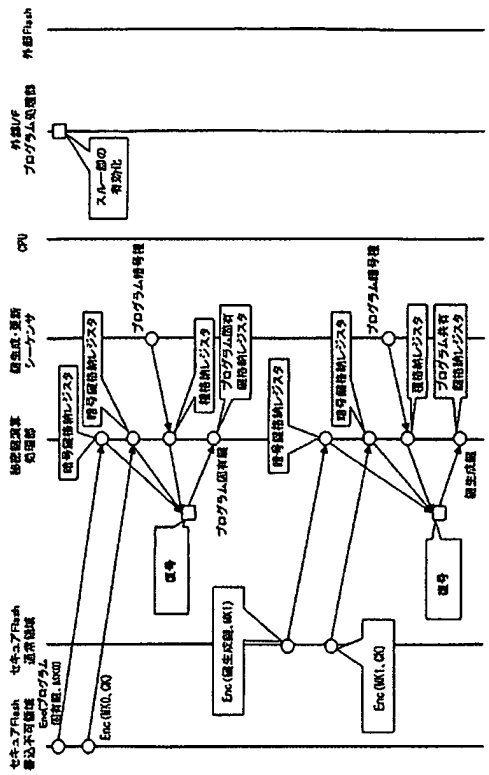
【図７】



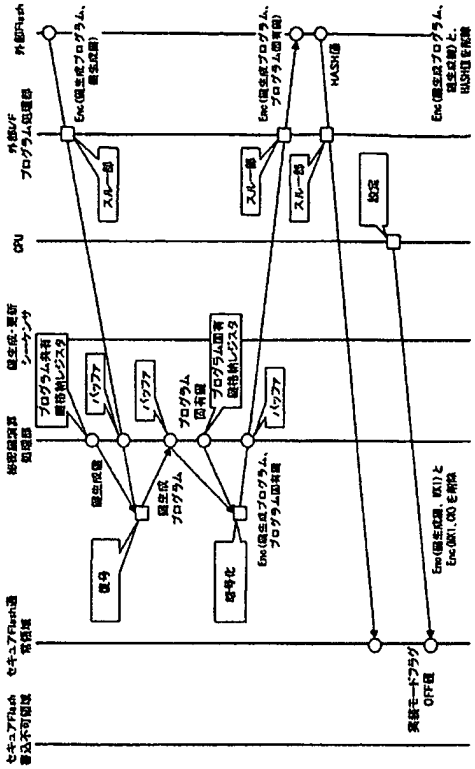
【図８】


SB1
開始
暗号化されたﾌﾟﾛｸﾞﾗﾑ固有鍵を設定
SB11
ﾌﾟﾛｸﾞﾗﾑ暗号種を用いてﾌﾟﾛｸﾞﾗﾑ固有鍵
を復号
SB12
暗号化された鍵生成鍵を設定
SB13
ﾌﾟﾛｸﾞﾗﾑ暗号種を用いて鍵生成鍵
を復号
SB14
外部Flash上の鍵生成鍵で暗号化された
鍵生成ﾌﾟﾛｸﾞﾗﾑを取り込み
SB15
鍵生成鍵で復号化し、ﾌﾟﾛｸﾞﾗﾑ固有鍵で
暗号化
SB16
ﾌﾟﾛｸﾞﾗﾑ固有鍵で暗号化された
鍵生成ﾌﾟﾛｸﾞﾗﾑを外部Flashに書き込み
SB17
外部Flash上のﾊｯｼｭ値を設定
SB18
実装ﾓｰﾄﾞをOFFに設定
SB19
ｾｷｭｱFlash上の暗号化された鍵生成鍵
を削除
SB1A
外部Flash上の鍵生成鍵で暗号化された
鍵生成ﾌﾟﾛｸﾞﾗﾑとHASH値を削除
SB1B
終了

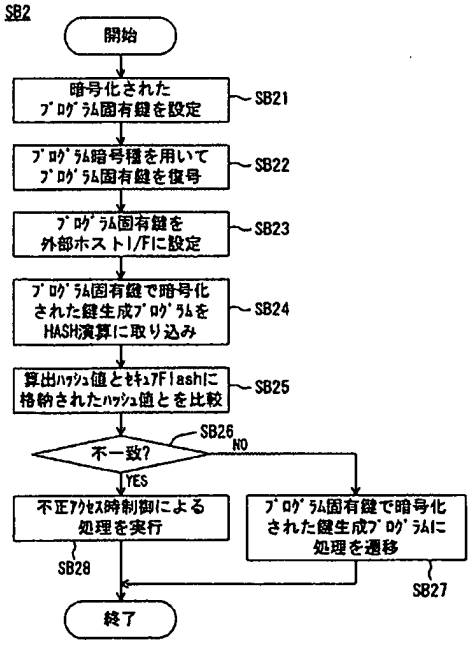
【図９】



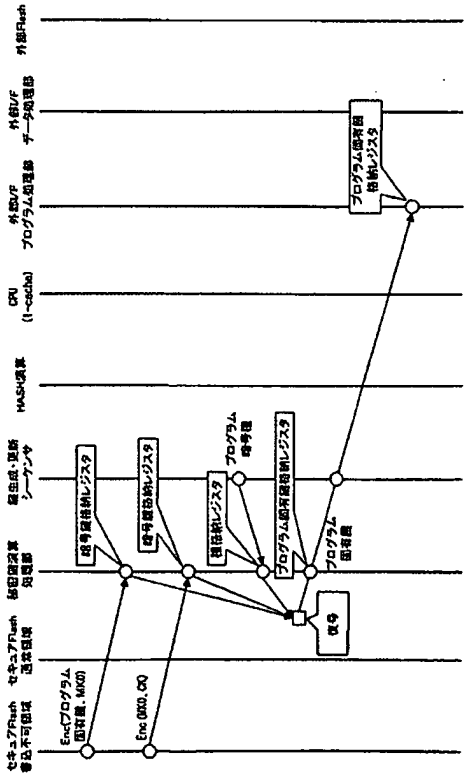
【図１０】



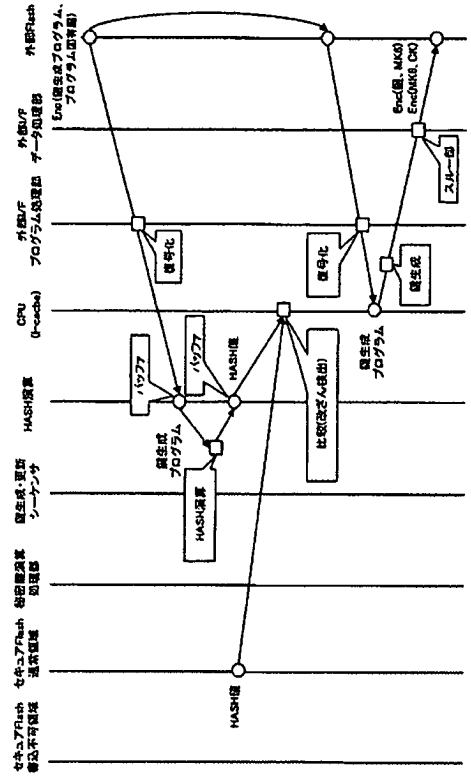
【図１１】



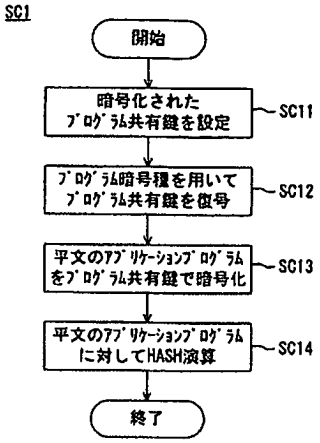
【図１２】



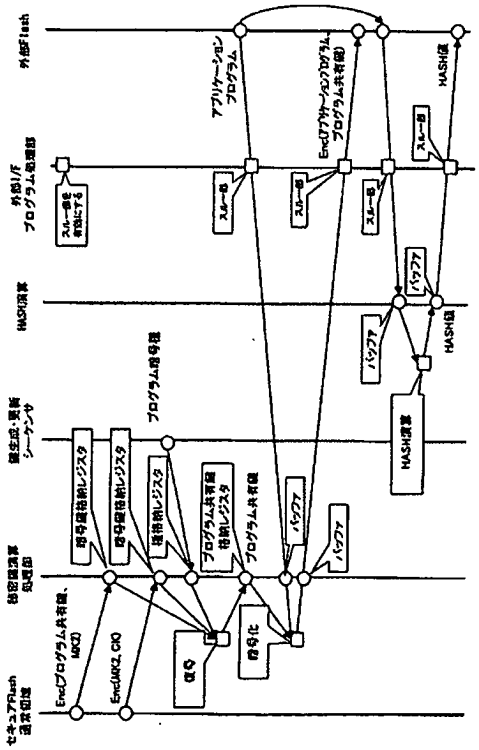
【図１３】



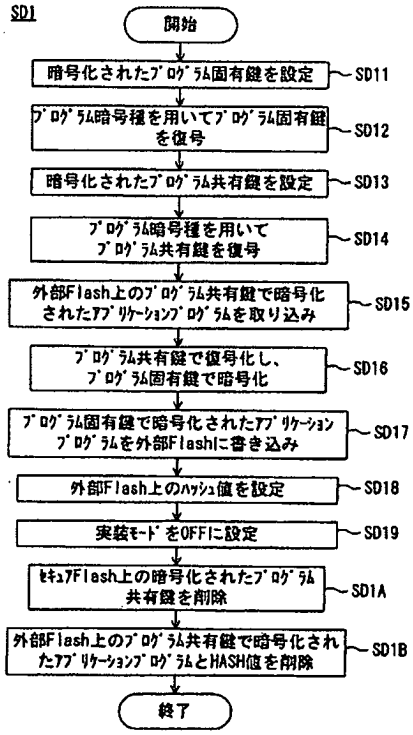
【図１４】



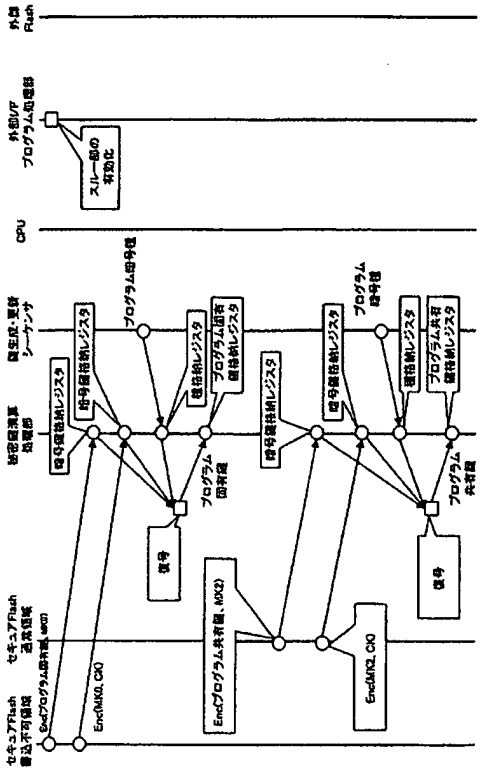
【図１５】



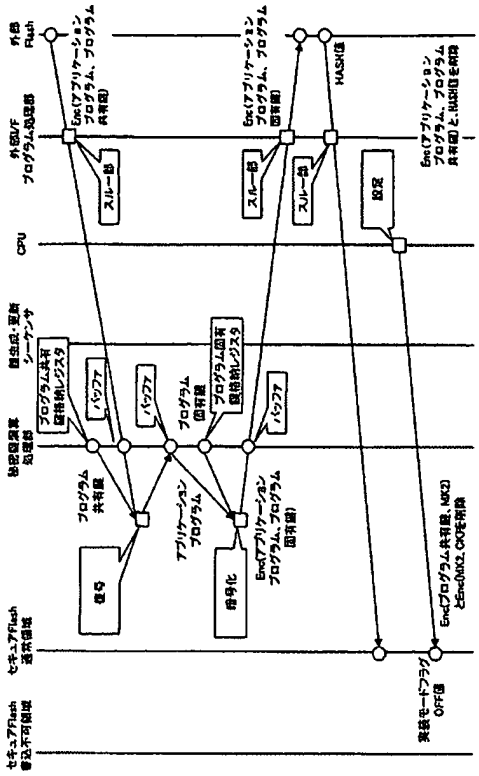
【図１６】



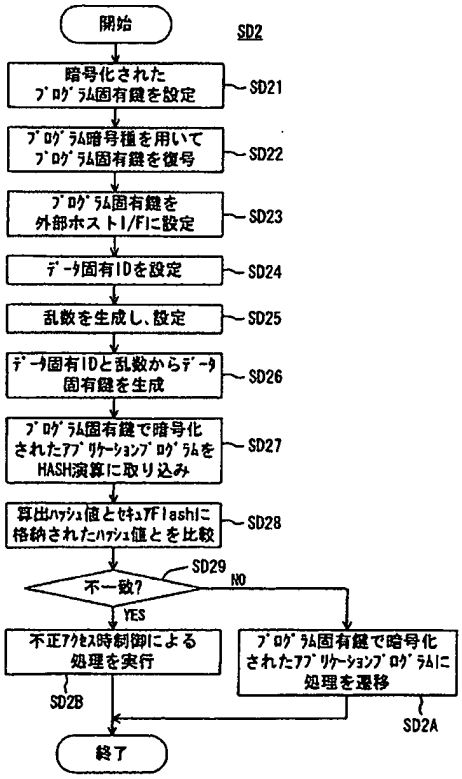
【図１７】



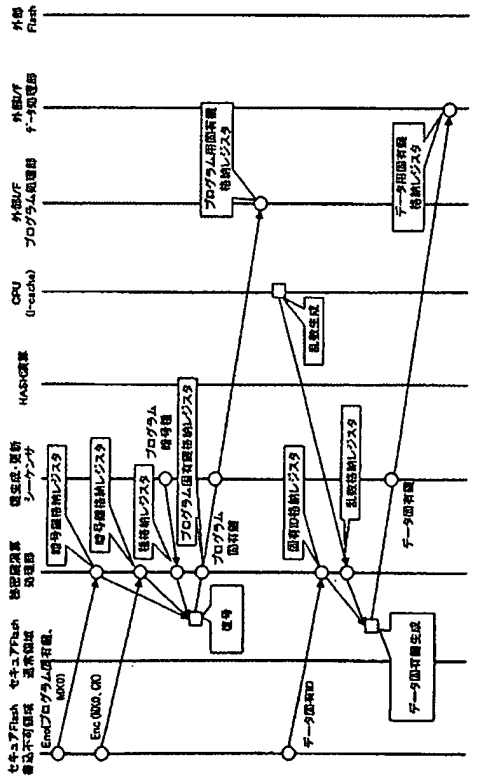
【図１８】



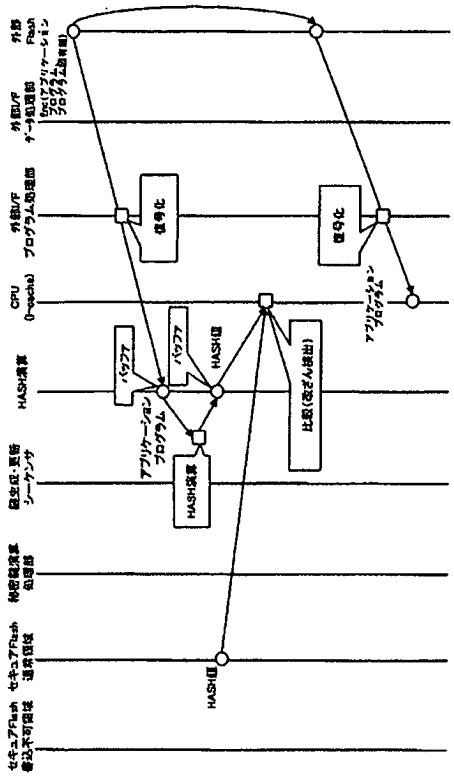
【図１９】



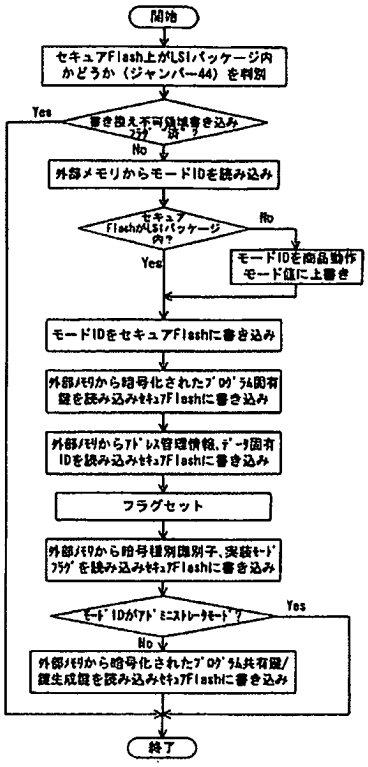
【図２０】



【図２１】


外部Flash
外部I/F データ処理部
外部I/F プログラム処理部
CPU (I-cache)
HASH演算
暗号化
復号化
バッファ
バッファ
HASH値
アプリケーションプログラム
HASH演算
比較(改ざん検出)
インターラプト
HASH値

【図２２】


SZ1
開始
セキュアFlash上がLSIパッケージ内かどうか（ジャンパー44）を判別
Yes
No
外部メモリからモードIDを読み込み
No
Yes
モードIDを商品動作モード値に上書き
モードIDをセキュアFlashに書き込み
外部メモリから暗号化されたﾌﾟﾛｸﾞﾗﾑ固有鍵を読み込みｾｷｭｱFlashに書き込み
外部メモリからｱﾄﾞﾚｽ管理情報、ﾃﾞｰﾀ固有IDを読み込みｾｷｭｱFlashに書き込み
フラグセット
外部メモリから暗号種別識別子、実装ﾓｰﾄﾞﾌﾗｸﾞを読み込みｾｷｭｱFlashに書き込み
Yes
No
外部メモリから暗号化されたﾌﾟﾛｸﾞﾗﾑ共有鍵/鍵生成鍵を読み込みｾｷｭｱFlashに書き込み
終了

フロントページの続き

(74)代理人 100115510
弁理士 手島 勝
(74)代理人 100115691
弁理士 藤田 篤史
(72)発明者 藤原 睦
大阪府門真市大字門真１００６番地 松下電器産業株式会社内
(72)発明者 根本 祐輔
大阪府門真市大字門真１００６番地 松下電器産業株式会社内
(72)発明者 安井 純一
大阪府門真市大字門真１００６番地 松下電器産業株式会社内
(72)発明者 前田 卓治
大阪府門真市大字門真１００６番地 松下電器産業株式会社内
(72)発明者 伊藤 孝幸
大阪府門真市大字門真１００６番地 松下電器産業株式会社内
(72)発明者 山田 泰司
大阪府門真市大字門真１００６番地 松下電器産業株式会社内
(72)発明者 井上 信治
大阪府門真市大字門真１００６番地 松下電器産業株式会社内
Fターム(参考) 5B076 FA01
5J104 AA12 PA14

【公報種別】特許法第１７条の２の規定による補正の掲載
【部門区分】第６部門第３区分
【発行日】平成17年4月7日(2005.4.7)

【公開番号】特開2004-54834(P2004-54834A)
【公開日】平成16年2月19日(2004.2.19)
【年通号数】公開・登録公報2004-007
【出願番号】特願2002-215096(P2002-215096)
【国際特許分類第７版】
　Ｇ０６Ｆ　1/00
　Ｇ０９Ｃ　1/00
【ＦＩ】
　Ｇ０６Ｆ　9/06　６６０Ｌ
　Ｇ０９Ｃ　1/00　６６０Ｄ

【手続補正書】
【提出日】平成16年5月28日(2004.5.28)
【手続補正１】
【補正対象書類名】明細書
【補正対象項目名】請求項５
【補正方法】変更
【補正の内容】
【請求項５】
　請求項１において、
　前記ＬＳＩと構成が共通のＬＳＩを、鍵生成用ＬＳＩとして、開発モードおよび商品動作モードとは異なる鍵生成モードに設定する工程と、
　前記鍵生成用ＬＳＩに、暗号化された鍵生成プログラムを実装し、この鍵生成プログラムを実行させることによって、鍵を生成する工程とを備えた
ことを特徴とするプログラム開発方法。
【手続補正２】
【補正対象書類名】明細書
【補正対象項目名】請求項７
【補正方法】変更
【補正の内容】
【請求項７】
　請求項５において、
　前記ＬＳＩと構成が共通のＬＳＩを、管理者用ＬＳＩとして、開発モード、商品動作モードおよび鍵生成モードとは異なるアドミニストレータモードに設定する工程と、
　前記管理者用ＬＳＩにおいて、前記鍵生成プログラムを開発し、任意の鍵で暗号化する工程とを備えた
ことを特徴とするプログラム開発方法。
【手続補正３】
【補正対象書類名】明細書
【補正対象項目名】００１０
【補正方法】変更
【補正の内容】
【００１０】
　請求項５の発明では、前記請求項１において、前記ＬＳＩと構成が共通のＬＳＩを鍵生成用ＬＳＩとして開発モードおよび商品動作モードとは異なる鍵生成モードに設定する工程と、前記鍵生成用ＬＳＩに暗号化された鍵生成プログラムを実装し、この鍵生成プログ

ラムを実行させることによって鍵を生成する工程とを備えたものとする。
【手続補正４】
【補正対象書類名】明細書
【補正対象項目名】００１２
【補正方法】変更
【補正の内容】
【００１２】
請求項７の発明では、前記請求項５において、前記ＬＳＩと構成が共通のＬＳＩを管理者用ＬＳＩとして、開発モード、商品動作モードおよび鍵生成モードとは異なるアドミニストレータモードに設定する工程と、前記管理者用ＬＳＩにおいて前記鍵生成プログラムを開発し、任意の鍵で暗号化する工程とを備えたものとする。